<<Project:Tank>>

Software Design Specification

Version 1.0

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 24 Juni 2016 | 1.0 | SDS pertama | Mananda, Winadi |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. **INTRODUCTION** 4
   1. PURPOSE 4
   2. SCOPE 4
   3. DOCUMENT’S INTENDED AUDIENCE 4
   4. OVERALL OVERVIEW 4
2. **SYSTEM OVERVIEW** 5
   1. MAJOR DESIGN ATIFACTS 5
   2. FUNCTIONAL REQUIREMENT 5
   3. SUPPORTING MATERIALS 6
   4. DEFINITION AND ACRONYMS 8
3. **DESIGN CONSIDERATIONS** 8
   1. ASSUMPTION 8
   2. CONSTRAINTS 8
   3. SYSTEM ENVIRONMENT 9
   4. DESIGN METHODOLOGY 9
   5. ANTICIPATION DURING DEVELOPMENT 10
   6. RISK AND FUTURE DESIGN CHANGES 10
4. **ARCHITECURAL DESIGN**
5. **HIGH LEVEL DESIGN**
6. **LOW LEVEL DESIGN**
7. **USER INTERFACE**

**LESSON LEARNED**

1. **Introduction**
   1. Purpose

Tujuan dari *Sofware Design Specification* secara umum adalah untuk menjelaskan bagaimana mendesain suatu sistem ditinjau dari beberapa segi, yaitu arsitektur sistem, high level maupun low level, dan beberapa pertimbangan desain lainnya yang perlu diperhatikan agar sistem yang dirancang dapat berjalan dengan semestinya.

Tujuan dari game Project:Tank adalah bagaimana cara implementasi gerak parabola pada sebuah game, dimana pada game Project:Tank gerak parabola akan diimplementasikan pada peluru yang ditembakkan

* 1. Scope

Pembangunan system *Project:Tank* akan dibuat dengan menggunakan *IDE NetBeans 8.1.* Adapun bahasa program yang akan digunakan adalah Java. Secara garis beras fitur-fitur utama yang diminta oleh *client* adalah game dengan memanfaatkan matematika serta fisika di dalamnya. Untuk lebih detailnya lihat bagian *product function* dan *software product features*.

* 1. Document’s Intended Audience

Dokumentasi ini ditujukan pada stakeholder yang berkepentingan dengan project ini

* 1. Overall Overview

Secara garis besar, game ini kami design sedemikian rupa sehingga dapat memenuhi kritetria penilain dalam mata kuliah matematika dan fisika untuk game. Beberapa design yang kami sudah buat adalah use case, pseudocode, class diagram, serta contoh UI yang direncanakan.

1. **SYSTEM OVERVIEW**
   1. **MAJOR DESIGN ARTIFACTS**

Artifacts yang utama darisistem ini adalah:

1. Software Design Spesification
2. Software Requirement Spesification
3. Coding Standard and Convention
4. **FUNCTIONAL REQUIREMENTS**

Dalam permainan Project:Tank, pemain akan diberi kendali atas sebuah target. Selama permainan berlangsung, akan ada tank yang menembak ke arah target. Tugas pemain adalah menghindari tembakan - tembakan dari tank sampai peluru tank habis dengan menggerakkan target ke atas atau ke bawah. Untuk menggerakkan target, pemain akan menggunakan tombol panah atas dan bawah pada keyboard. Dalam permainan, pemain akan diberi sejumlah nyawa, yang akan berkurang jika pemain terkena tembakan. Pemain akan kalah jika nyawa tersebut habis sebelum peluru tank habis. Pemain akan menang jika setelah peluru tank habis, nyawa pemain masih tersisa. Peluru tank terdiri dari 3 jenis:

● Regular, peluru normal tanpa modifikasi apapun

● Split, peluru akan terpecah menjadi 2 (atau lebih) peluru di udara

● Burst, 2 (atau lebih) peluru akan sekaligus ditembakkan dalam sekali tembak

Game ini akan memiliki 3 jenis tingkat kesulitan, yaitu easy, medium, dan hard. Tingkat kesulitan dibedakan atas beberapa factor seperti jumlah ammo, tingkat kecepatan peluru ditembakkan, jarak antar tank dan target, power dari peluru, dll.

1. **SUPPORTING MATERIALS**

-

1. **DEFINITION AND ACRONYMS**

**IDE:** Integrated Development Environement adalah aplikasi yang memberikan fasilitas untuk development software.

**HP:** Health Point

**RAM:** Random Access Memory, komponen hardware untuk penyimpanan memory sementara

**Cooldown:** waktu/delay yang dibutuhkan untuk skill dapat dipakai kembali

**Real-time Strategy:** genre game strategi, dengan waktu secara real-time

1. **DESIGN CONSIDERATIONS**
   1. **ASSUMPTION**

Berikut merupakan asumsi yang kami buat dalam pembuatan game Project:Tank:

Mengerjakan sistem pergerakan tank menjadi prioritas sebelum mengimplementasi sistem peluru. Implementasi tersebut membutuhkan rumus Matematika dan Fisika. Maka, perlu dikumpulkan terlebih dahulu rumus yang ingin / mungkin digunakan dalam implementasi.

* 1. **CONSTRAINTS**

Terdapat beberapa batasan/pertimbangan dalam pembuatan desain (sesuai dengan yang terdokumentasi dalam SRS):

1. Pembuatan game ini tidak akan menggunakan game engine apapun, hanya menggunakan software Integrated Development Environment (IDE) dengan bahasa pemrograman Java.

2. Desain dan pembuatan dari amunisi beserta rumus - rumus yang akan digunakan menjadi prioritas dari pembuatan game ini. Pelengkap - pelengkap seperti aset tekstur dan suara akan ditambahkan setelah sistem amunisi penulis telah selesai

3. Salah satu mode dalam game ini dimainkan oleh 2 orang, namun mode ini tidak dapat dimainkan secara multipemain lewat koneksi LAN/Internet.

* 1. **SYSTEM ENVIRONMENT**

Game ini akan didevelop menggunakan Java, sehingga game juga akan dijalankan menggunakan Java

* 1. **DESIGN METHODOLOGY**

UML Approach Design: Menggunakan UML untuk mendeskripsikan semua requirements. UML diagram yang digunakan antara lain: Use case diagram, collaboration, activity dan deployment diagram.

* 1. **ANTICIPATION DURING DEVELOPMENT**

Antisipasi yang dilakukan bila ada perubahan desain selama proses pembuatan game Project:Tank

-Mengkosultasikan dahulu dengan dosen terkait sebelum menerapkan perubahan

-Merubah class yang terkena dampak perubahan desain

* 1. **RISK AND FUTURE DESIGN CHANGES**

Karena adanya limitasi waktu, maka *game* yang kami buat ini tidaklah sempurna, karena itu ada beberapa resiko yang harus diantisipasi, yaitu:

* Resiko keamanan

Game yang kami buat tidak menerapkan security pada gamenya, sehingga mungkin saja terdapat cheat yang merusak game

* Resiko bug

Mungkin saja muncul bug-bug yang tidak terdeteksi pada saat testing

Game yang kami buat memiliki potensi untuk dikembangkan menjadi lebih sempurna, antara lain:

1. Menambal bug-bug yang muncul pada permainan
2. Menambah fitur-fitur baru dalam game
3. Memperbaiki tampilan game dengan menggunakan aset - aset yang lebih beragam, tekstur, efek visual, dsb.
4. Memperluas mekanika game

1. **ARCHITECTURAL DESIGN**

**World**

**Tank**

**Target**

**Bullet**

Diagram di atas menjelaskan relasi antar class pada game Project:Tank. World merupakan class yang membuat object dari class lainnya untuk ditampilkan pada JFrame.

1. **HIGH LEVEL DESIGN**

**1. Introduction**

**1.1. Purpose**

Bagian High Level Design bertujuan untuk memberikan gambaran dari game yang kami buat dari sudut pandang yang memungkinkan bagi pembaca di luar bidang teknis untuk dapat memahaminya. Penjelasan akan dilakukan dengan menggunakan istilah – istilah yang hampir non-teknis dengan harapan dapat menjelaskan secara high-level.

**2. General Description**

**2.1. Product Perspective**

Game ini kami buat dengan menggunakan konsep Object-Oriented Programming (OOP). Konsep ini fokus pada setiap objek / entitas yang ada, dan memisahkannya menjadi class – class dengan atribut dan fungsi yang relevan. Dengan pertimbangan ini, game kami akan terdiri dari (kurang lebih) 5 class, yaitu Main, World, Bullet, Tank, dan Target (arsitektur akan ditunjukkan pada bab selanjutnya) dengan World sebagai class yang akan menampung setiap instansiasi dari class lainnya. Bahasa yang digunakan untuk pembuatan game ini adalah Java. Sebagian dari game ini akan kami program sendiri, sebagian lagi kami akan melakukan reuse dari sumber lain dan melakukan penyesuaian ketika diimplementasi ke dalam code.

**2.2. Tools Used**

**1.** Java IDE (Eclipse, NetBeans, dll.)

**2.3 General Constraints**

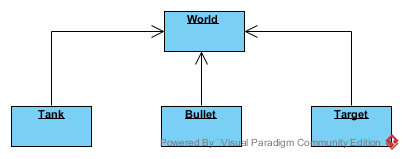
Game ini dibuat tanpa menggunakan game engine / game maker apapun. Di dalam game juga diharuskan adanya implementasi Matematika dan Fisika sesuai dengan requirements yang telah ditetapkan pada awal projek.

**2.4 Assumptions**

Game ini dibuat dengan asumsi bahwa dengan ide yang telah kami tentukan, kami dapat memenuhi semua requirements yang ada. Asumsi lain adalah kami diperbolehkan untuk melakukan reuse pada bagian – bagian tertentu untuk menghemat waktu dan usaha.

**3. Design Details**

**3.1 Application Architecture**



**3.2 User Interface**

User Interface dari game ini akan dibuat sederhana. Di dalamnya akan ditampilkan informasi – informasi penting untuk pemain dan menu untuk diakses serta representasi tank, peluru, dan target.

**3.3 Maintainability**

Untuk game yang kami buat untuk projek ini tidak akan ada maintenance yang dilakukan, mengingat game ini hanya untuk keperluan memenuhi tugas projek saja. Jika ada pertimbangan lain, mungkin maintenance akan dipertimbangkan

1. **LOW LEVEL DESIGN**

Beberapa *source code* penting yang digunakan:

**Laju Peluru di Udara**

public void move(double t)

{

double currentT = t - t0;

currentX = x0 + (getV0x()\*currentT+ getAx()\*currentT\*currentT/2);

currentY = y0 + (getV0y()\*currentT+ getAy()\*currentT\*currentT/2);

}

**Memutar Barrel Tank**

AffineTransform at = new AffineTransform();

at.translate(this.getWidth() / 2, this.getHeight() / 2);

at.rotate(tank.getAlpha());

at.scale(0.4, 0.4);

at.translate(barrelKanan.getWidth(), barrelKanan.getHeight());

tank.drawBarrel2D(g2d, this.barrelKanan, at);

**Mencetak Graphic ke Layar**

public void printScreen()

{

//Tampilkan ke screen

try

{

Graphics g = getGraphics();

if(dbImage != null && g != null)

{

g.drawImage(dbImage, 0, canvasStartY, null);

}

// Sync the display on some systems.

// (on Linux, this fixes event queue problems)

Toolkit.getDefaultToolkit().sync();

g.dispose();

}

catch(Exception ex)

{

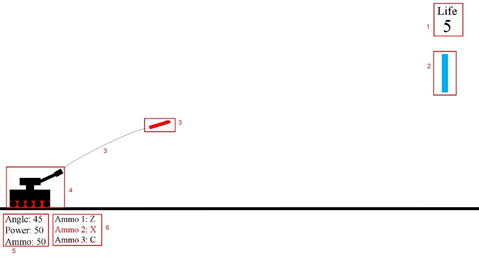
System.out.println("Graphics error: " + ex);

}

}

1. **USER INTERFACE**

**Easy**



Berikut adalah penjelasan gambar di atas sesuai nomor yang tertera di gambar.

1. Nyawa: tampilan tersebut menunjukkan status nyawa dari pemain. Jika terkena peluru maka akan berkurang 1

2. Target: target yang digerakkan pemain akan berada di area tersebut, digerakkan atas bawah

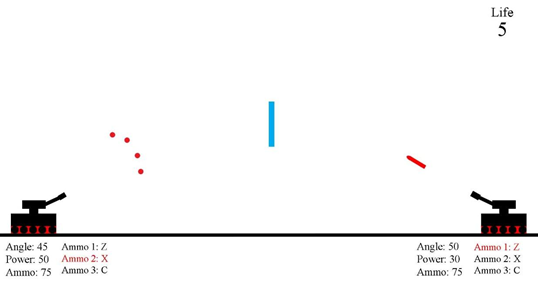
3. Peluru: peluru yang ditembakkan oleh tank, dan terdapat garis yang menunjukkan lintasan peluru

4. Tank: tank yang menembakkan peluru, dengan random angle larasnya

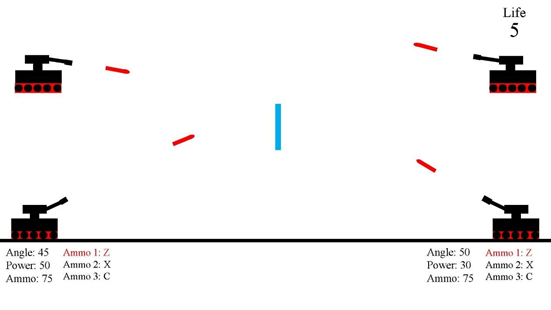
5. Status: menunjukkan angle, power (kekuatan peluru ditembakkan), serta ammo yang tersisa. Jika ammo menyentuh angka 0 maka game akan selesai.

6. Ammo: jenis ammo yang sedang ditembakkan, dan akan random jenisnya per tembakkan

**Medium**



**Hard**



Tampilan tingkat kesulitan di atas hanya berupa rancangan design level, dan dalam implementasinya bisa berbeda dalam hal tampillannya, jarak antar tank dan target, ukuran dari target, penempatan tulisan, dll.

**9. LESSON LEARNED**

Dalam pembuatan dokumen Software Design Spesification, kami belajar beberapa hal mengenai bagaimana cara membuat dokumen Software Design Spesification yang baik, bagaimana mendesign baik high-level maupun low-level untuk sebuah project, serta menambah wawasan baru mengenai dokumen yang harus dibuat dalam sebuah project.